Les étapes principales de la création d'un corpus bilingue aligné

Ágoston NAGY

Introduction

L'alignement ou l'appariement automatique de deux textes consiste à mettre en correspondance chaque unité logique de l'un des textes avec une unité de l'autre qui en comprend la traduction. Dans la plupart des cas, ce sont les phrases ou les mots qui sont alignés ou appariés, mais il est également possible de mettre en correspondance des syntagmes. Cependant, ce qui est le plus courant et utile, c'est l'alignement de phrases, c'est-à-dire la mise en correspondance d'une (ou de plusieurs) phrase(s) de la langue-source avec l'unité qui y correspond dans la langue-cible.

Ces corpus alignés, appelés également *bitextes*, sont réalisés par des systèmes d'alignement, et ils sont appréciés, avant tout par les traducteurs qui peuvent y retrouver le pendant d'un terme ou d'une expression dans une autre langue. Les bitextes ont également une énorme utilité pour les ordinateurs mêmes qui les créent pour établir des dictionnaires bilingues de probabilité (à préciser dans la section 3 de cet article) lesquels sont exploités par les traducteurs automatiques. (Langlais et Véronis 1999)

Aligner un corpus bilingue se passe en deux étapes : la première consiste à segmenter les deux textes en phrases et la deuxième sert à aligner chaque phrase du texte d'origine avec une unité (qui n'est pas toujours une phrase) du texte traduit. Le but de cet article est de démontrer que la segmentation en phrases et l'alignement ne sont pas aussi évidents (du moins pour l'ordinateur) qu'il ne le paraît à première vue.

Pour présenter les difficultés qui peuvent surgir durant la première étape, c'est-à-dire la segmentation, nous ferons allusion aux résultats que nous avons expérimentés au cours des exécutions d'un segmenteur dont le code a été écrit par nous-même, et dans cet article nous décrirons les étapes de la modification de ce code. Ce segmenteur a été réalisé en Perl (langage de programmation adapté pour Linux mais également compatible avec MS-DOS), et a été exécuté sur deux textes littéraires dont l'un est la traduction de l'autre. Le texte d'origine était le premier livre du roman intitulé *Le rouge et le noir* (*RN* par la suite) de Stendhal, le second en était la traduction hongroise (*VF* par la suite)[1].

[1] Les deux textes sont accessibles sur le Web à l'adresse suivante :
STENDHAL, *Le rouge et le noir*, 1830 [http://abu.cnam.fr/cgi-bin/donner?rouge1] (cette édition numérisée est basée sur l'édition du Paris, Livre Club du Libraire, 1959, prétendue conforme à l'original) (date de la consultation : le 10 novembre 2006)
STENDHAL, *Vörös és fekete*, (Traduction : Endre Illés),
[http://www.mek.iif.hu/porta/szint/human/szepirod/kulfoldi/stendhal/vorosf/] (cette édition a été

1. Les problématiques de la phrase

Les corpus bilingues sont, par convention, alignés au niveau de phrases. Mais pourquoi au niveau de phrases si l'alignement peut être effectué au niveau de paragraphes aussi (ce qui serait d'ailleurs plus simple) ? D'une part, parce que les segments plus volumineux sont difficiles à gérer et pour les traducteurs, un corpus aligné au niveau de paragraphes est moins utile, le temps consacré à la recherche étant plus important. D'autre part, c'est au niveau de la phrase que se construit la prédication (c'est-à-dire l'adjonction du sujet au prédicat) – c'est donc le segment de base que la linguistique informatique a hérité des grammaires (Fuchs 1993). Cependant, la segmentation en phrases pose des difficultés et pour les humains et pour l'ordinateur, ainsi le but de cette section est d'en présenter les différents points de vue.

Les grammaires traditionnelles (Arrivé et al. 1986 ou Riegel et al. 1994) se réfèrent toujours à trois critères de base qui permettent d'identifier les phrases dans un texte complet. Le premier est le critère graphique qui suppose que toute phrase est une suite de mots délimitée par une majuscule au début et par une ponctuation finale à la fin. Côté phonétique, une phrase est une séquence de phonèmes délimitée par deux pauses (une au début et une à la fin) et elle est marquée par une intonation caractéristique au type de la phrase donnée (interrogative, affirmative ou impérative). Selon le troisième critère, plutôt sémantique, la phrase constitue toujours une « unité de pensée », par conséquent tout assemblage de mots n'est pas toujours une phrase.

Les contre-exemples et les contradictions étant nombreux, les grammairiens n'en profitent pas moins pour s'attaquer à ce sujet assez problématique. Toutefois il est à reconnaître que ces trois critères sont insuffisants. En premier lieu, pour démontrer que la ponctuation ne suffit pas pour délimiter les phrases, nous pouvons observer que le point d'exclamation ou le point d'interrogation ne marquent pas toujours la fin de la phrase, ce qui peut être illustré par l'exemple suivant :

(1) Eh! madame, vous serait-il arrivé quelque malheur? (RN, Chapitre 7)

Ensuite, des pauses peuvent surgir à l'intérieur de la phrase même, comme dans le cas des constructions disloquées, par exemple :

(2) Marie, le vin, elle l'aime.

Et finalement qu'entendre par « unité de pensée » ? La « complétude sémantique » ne saurait pas toujours déterminer si telle ou telle séquence de mots constitue une ou plusieurs phrases. Si *Il se fait tard. Marie veut partir.* constitue deux phrases, donc deux unités de pensée, alors pourquoi la chaîne *Il se fait tard, Marie veut donc partir.* n'en constitue-t-elle qu'une seule ?

numérisée par l'éditeur Interpopulart Könyvkiadó, à Szentendre) (date de la consultation : le 10 novembre 2006)

Nous pouvons donc constater que la notion de la phrase n'est pas si évidente qu'il ne le semble à première vue. Dans cet article, nous recourrons au premier critère de base, d'ordre graphique, pour délimiter les phrases dans un texte donné ; cependant, nous le modifierons et préciserons parce qu'une lettre majuscule initiale et une ponctuation finale ne suffiront pas toujours pour délimiter les phrases. Le deuxième critère n'est pas applicable dans notre cas, comme notre corpus est écrit et ne contient aucune indication phonétique. Le troisième critère est également problématique puisque les applications simples relevant du domaine de la linguistique informatique (comme les systèmes d'alignement) n'ont pas été créées pour pouvoir traiter le sens, et la notion d'« unité de pensée » est difficile à interpréter même pour les humains. En plus, son implémentation informatique dépend de la langue du texte et doit prendre beaucoup plus de temps.

2. Tentatives et problématiques de segmentation sur la base du critère typographique

Les locuteurs d'une langue pensent que la segmentation de texte en phrases est une tâche relativement facile, car il suffit de prendre en considération les majuscules et la ponctuation. Malheureusement, la vérité semble contredire cette théorie dans la plupart des cas. Par exemple, dans le corpus Brown, rien que 90% des points marquent la frontière d'une phrase, les 10% qui restent apparaissent dans les abréviations, et dans des nombres fractionnaires (Gale et Church 1993).

Par la suite, nous reverrons quelques problèmes à résoudre par les segmenteurs, nous concentrant essentiellement sur les difficultés que posent les sigles et les autres abréviations. Nous ferons également allusion à des phénomènes qui n'ont pas apparu dans les textes donnés mais qui peuvent surgir à tout moment ailleurs.

Le plus grand problème pour les segmenteurs a toujours été provoqué par les abréviations et les sigles. Ceux-ci constituent un danger pour ces programmes parce que le point qui se trouve à la fin ou même à l'intérieur du sigle ne peut nullement être une frontière de phrase (ou il ne l'est que si le sigle se trouve vraiment à la fin du texte). Dans la version française du roman, il n'y a qu'une seule abréviation : le *M.* pour *Monsieur*. Ce qui est problématique dans le cas de cette abréviation, c'est que le point est suivi par une majuscule qui ne marque pas le début d'une nouvelle phrase, par exemple *M. Ducrest*. La solution qui se pose pour pallier ce problème est d'exclure le caractère *M* avant les signes de ponctuation finaux. Mais si nous considérons un texte quotidien, accessible à travers le Web ou la presse, nous pouvons vite nous rendre compte que notre code doit être encore amélioré. Regardons l'extrait suivant :

(3) La grève à la S.N.C.F. a perturbé le trafic mercredi.

A partir de cet exemple, nous pouvons constater que les contraintes précédentes ne suffiront pas. Pour résoudre ce problème, il pourrait être utile d'exclure les majuscules avant les signes de ponctuation finaux ; de cette façon, la plupart des

problèmes seraient filtrés, mais si l'entrée d'une application TAO (Traduction assistée par ordinateur) n'est constituée que par des majuscules, alors le logiciel devrait-il le traiter d'une seule unité ?

Il vaudrait mieux prescrire que la ponctuation finale doit toujours être suivi d'un espace ou d'un signal de fin de ligne (ce dernier est '\n' en C, et '$' en Perl), comme, par convention, aucun espace n'est laissé après les points se trouvant dans les sigles et qu'un espace soit inséré après le point final ou après d'autres signes de ponctuation (par exemple ; ,). Cependant, cette méthode peut également aboutir à des malfonctionnements si un espace a été involontairement laissé tomber ou inséré (même si la probabilité en est assez petite).

Pour que le segmenteur fonctionne d'une façon adéquate, la plupart des applications recourent à un dictionnaire d'abréviations et de sigles à l'aide duquel le logiciel peut vérifier si tel ou tel point fait partie d'un sigle ou marque la fin d'une phrase (Mikheev 2003). Cependant, ceux qui connaissent les traditions françaises savent bien que – depuis le latin – une grande quantité de sigles voient le jour à chaque instant, et même s'il est possible de stocker tous ces mots courts dans un dictionnaire, la base de données devrait être réactualisée chaque semaine. Il suffit de penser au C.P.E. (Contrat Première Embauche), qui, pendant quelques jours, a envahi toute la presse française l'année dernière.

Comme la segmentation précise en phrases est souvent problématique, plusieurs méthodes ont déjà été développées pour réaliser ce but. Pour se référer à ce phénomène, les articles scientifiques ont recours au terme SBD (Sentence Boundary Disambiguation – Désambiguïsation de frontière de phrase). Il existe en principe deux types d'applications SBD : le premier type comprend les applications à base de règles, l'autre comprend des applications qui fonctionnent par des méthodes statistiques (Mikheev 2003).

Les systèmes à base de règles ont recours à des règles rédigées à la main (à des expressions régulières), lesquelles sont complétées par des dictionnaires de sigles et de noms propres. Notre système est également un système à base de règles : une telle application est facile à établir, mais la perfectionner est une tâche beaucoup plus difficile et prend bien du temps.

Un tel segmenteur a été créé par Anne Dister (Université de Liège) dans l'interface INTEX-NOOJ, ou par exemple y appartient le système SATZ, qui a également pris en considération le contexte gauche et droit des candidats définissant les fins de phrases, il a donc intégré un module lexical dans son système (Mourad 1999).

Les outils statistiques sont basés sur le fait que l'application elle-même est capable d'apprendre des règles à la base des corpus pré-segmentés, toutefois, il existe aujourd'hui des systèmes qui peuvent extraire des règles à partir des textes non-segmentés préalablement. Ces systèmes tirent avantage de la fréquence relative de la régularité de la ponctuation et du nombre moins considérable des irrégularités pouvant éveiller les soupçons. Un tel système est par exemple, LTSTOP. Ces segmenteurs sont plus efficaces, et sont adaptables à plusieurs langues. Le taux

d'erreur de ces applications est moins considérable que celui des modèles à base de règles. (Mikheev 2003)

3. Alignement de textes bilingues

La segmentation en phrases n'est pas profitable en elle-même parce qu'elle n'est que le premier pas pour la plupart des logiciels manipulant la langue. Dans le cas où il existe deux textes segmentés dont l'un est la traduction de l'autre, il serait utile de les aligner en mettant en correspondance les phrases du texte original avec leurs traductions car il est communément admis que l'aide la plus importante qu'un traducteur puisse avoir est l'accès à un corpus de traductions précédentes. Toutefois un corpus bilingue n'est utile que si le correspondant d'un élément de la langue-source peut être vite localisé dans le texte de la langue-cible. Ceci est plus simple si les deux textes ont été alignés auparavant (Kay & Röscheisen 1993).

L'alignement est également un processus nécessaire pour établir un dictionnaire bilingue de probabilité, lequel constitue l'une des bases principales pour les traducteurs automatiques fonctionnant à l'aide des méthodes statistiques. Les applications de cette sorte se créent des dictionnaires bilingues à l'aide d'un grand corpus bilingue aligné. Elles mettent en correspondance chaque mot du texte original avec un ou plusieurs mots du texte cible, elles alignent donc les deux textes au niveau du lexique et elles attachent une valeur de probabilité au correspondant-candidat du mot donné. Les dictionnaires établis de cette façon serviront de base à la traduction automatique dans l'avenir. Le tableau suivant montre l'entrée du déterminant anglais *the* dans un tel dictionnaire bilingue :

Français	Probabilité
le	.610
la	.178
l'	.083
les	.023
ce	.013
il	.012
de	.009
à	.007
que	.007

Tableau 1 Probabilités pour « the »

Il n'est pas surprenant du tout que l'article défini anglais soit traduit dans la plupart des cas en *le* ou *la*, ce qui est frappant, c'est que c'est un logiciel qui a abouti à un tel résultat. Le tableau nous montre également que le moins probable est que le correspondant-candidat du mot *the* est en français *que* ou *de*.

3.1. Problématiques de l'alignement

L'alignement, tout comme la segmentation, est un processus qui ne va pas de soi. Cette tâche est souvent rendue difficile par le traducteur qui ne respecte pas toujours la segmentation du texte d'origine – surtout pour des effets stylistiques. Dans le cas où une phrase est traduite par une seule phrase, (ce qui est le cas le plus fréquent) il n'y a pas de problème, mais si une seule phrase est traduite par deux ou trois phrases, l'alignement de ces segments pose des difficultés. Dans le cas des deux romans analysés, le traducteur a souvent opté pour des phrases moins longues que celles d'origine, ce qui est bien illustré par (4):

> (4) Leur croissance rapide et leur belle verdure tirant sur le bleu, ils la doivent à la terre rapportée, que M. le maire a fait placer derrière son immense mur de soutènement, car, malgré l'opposition du conseil municipal, il a élargi la promenade de plus de six pieds (quoiqu'il soit ultra et moi libéral, je l'en loue), c'est pourquoi dans son opinion et dans celle de M. Valenod, l'heureux directeur du dépôt de mendicité de Verrières, cette terrasse peut soutenir la comparaison avec celle de Saint-Germain-en-Laye. (RN, Chapitre 2)
>
> Gyors növekedésüket, kékes árnyalatú, szép zöldjüket annak a földtömegnek köszönhetik, amit de Rênal úr hordatott a nagy támaszfal mögé. A városi tanács ugyan ellenezte a sétány megnagyobbítását, ő mégis hatlábnyival kiszélesítette, s ezért megérdemli dicséretünket (bár a polgármester úr *ultra,* én meg *liberális* vagyok). Véleményem szerint a sétány bátran összehasonlítható Saint-Germain-en-Laye teraszaival, s ítéletét még Valenod úr, a verrières-i szegényház szerencsés igazgatója is osztotta.

La phrase française (assez longue) en (4) a été traduite en hongrois en trois phrases distinctes. Dans ce cas-là, ces trois phrases devraient être considérées comme une unité logique au cours de l'alignement car il ne serait pas logique de trancher la phrase française entre *soutènement*, et *car* pour pouvoir aligner chaque phrase hongroise à une phrase française.

Des systèmes à base de règles ne seront pas capable d'aligner des unités pareilles comme il n'existe pas de règles qui peuvent prescrire précisément quand une phrase peut être appariée à une ou plusieurs phrases. Les logiciels d'alignement automatiques ont donc recours plutôt aux méthodes statistiques.

3.2. Systèmes d'alignement de textes bilingues

Dans cette sous-section nous présenterons deux outils d'alignement automatiques, celui de Gale et Church (1993) et celui de Kay et Röscheisen (1993), ces derniers étant considérés comme les plus importants et les plus significatifs dans ce domaine. Même si ceux-ci fonctionnent à l'aide des méthodes statistiques, il existe une différence considérable entre les deux : le premier tente d'aligner deux textes à la base du nombre des caractères, le second à partir des mots qui s'y trouvent.

Le programme *align* de Gale et Church (1993) est basé sur un modèle très simple : le nombre des caractères d'une unité logique est souvent plus au moins équivalent à celui de sa traduction, c'est-à-dire les phrases longues ont tendance à

être traduites comme des phrases longues, et c'est l'envers pour les phrases courtes. Une valeur de probabilité est alors attachée à chaque couple de phrases qui peuvent se correspondre – et cette valeur est calculée à partir de la longueur des phrases (en caractères). Il est vraiment surprenant qu'une si simple méthode puisse marcher, cependant les résultats préalables en ont prouvé le succès. Une question évidente et logique se pose à ce point : qu'est-ce qui se passerait si l'alignement se faisait à la base du nombre des mots (car c'est les mots qui sont traduits plutôt que les caractères) ? Il est intéressant de noter que le premier livre de la version hongroise du RN comprenait 59527 mots, par contre, la version d'origine 78612, donc la différence est considérable, elle est de 32%. Cependant, si le nombre des caractères est choisi comme critère, la différence est moins énorme (351080 caractères dans la version traduite, 375793 dans la version originale, donc 7% de différence). Il est donc raisonnable de s'intéresser plutôt au nombre des caractères.

La méthode de Kay et Röscheisen (1993) relève également du domaine de la statistique mais elle est plutôt une méthode lexicale, ce qui peut sembler difficile à réaliser à première vue car il n'est pas toujours facile de faire correspondre un mot à un autre dans une version traduite. Cependant, il est relativement simple d'aligner des noms propres (qui ne sont jamais traduits) ou des termes techniques (qui sont toujours traduits de la même façon). En plus, cette méthode est adaptable pour n'importe quelle paire de langues. Pourtant, elle est extrêmement lente car elle se sert trop des ressources de l'ordinateur : en premier lieu, elle stocke le mot de la version d'origine avec le nombre des phrases où il apparaît, ensuite elle stocke également une liste de paires de mots (avec une valeur de probabilité) qui ont été alignés et qui sont considérés comme les traductions de l'un et de l'autre. C'est la raison pour laquelle ce dernier est très lent (en termes d'informatique) : il prend plusieurs heures pour aligner un article (Gale et Church 1993). Cependant, ce logiciel établit déjà un dictionnaire de probabilité à partir des deux versions du texte avec un minimum d'erreurs.

Conclusion

Dans cet article, nous avons énuméré et détaillé les deux étapes principales qui sont inévitables au cours de la création d'un corpus bilingue : la segmentation en phrases et l'alignement. Il a été constaté au début que ces deux processus ne sont pas si évidents (du moins pour l'ordinateur) et que chacun d'entre eux exige des précisions.

La segmentation en phrases est une étape qui peut être considérée comme un processus dont l'algorithme peut être réalisé de deux façons : d'une part par des règles écrites à la main, d'autre part par des méthodes statistiques. Les systèmes ayant recours à la première méthode (comme notre propre segmenteur) sont moins difficiles à créer mais ils sont plus difficiles à perfectionner, tandis que les segmenteurs du deuxième type sont plus difficiles à créer mais ils sont adaptables pour n'importe quelle langue avec un taux d'erreur de 1,5% au maximum.

Si une seule phrase était toujours traduite par une seule autre, l'appariement des textes ne poserait pas de problèmes pour les systèmes d'alignement mais comme ce n'est pas le cas, il faut trouver un algorithme adéquat pour ce processus. Contrairement à la segmentation, pour aligner des textes, il n'existe qu'une seule possibilité, notamment l'utilisation de méthodes statistiques. Dans cet article, nous avons présenté les deux méthodes les plus connues et les plus importantes des méthodes statistiques permettant d'aligner deux textes. Gale et Church (1993) se basent sur le fait que le nombre des caractères d'une phrase est plus ou moins équivalent à celui des caractères de la phrase qui y correspond. Par contre, Kay et Röscheisen (1993) tentent de faire correspondre deux phrases à partir des mots qui s'y trouvent.

Bibliographie

ARRIVE, Michel – GADET, Françoise, GALMICHE, Michel, *La grammaire d'aujourd'hui: Guide alphabétique de linguistique française,* Paris, Flammarion, 1986.

FUCHS, Catherine, Linguistique et Traitement automatique des langues, Paris, Hachette, 1993.

GALE, William A., CHURCH, Kenneth W., « A program for aligning sentences in bilingual corpora », *Computational Linguistics,* 19(3), 1993, p. 75-102.

KAY, Martin, RÖSCHEISEN, Martin, « Text-translation alignment », *Computational Linguistics,* 19(3), 1993, p. 121-142.

VERONIS, Jean, LANGLAIS, Philippe, *Évaluation de systèmes d'alignement de textes multilingues*, 1999, http://www.iro.umontreal.ca/~felipe/Papers/livrejst97.rtf (le 10 décembre 2006)

MIKHEEV, Andrei. « Text Segmentation », in : MITKOV, Ruslan, ed. *The Oxford Handbook of Computational Linguistics*, Oxford, Oxford University Press, 2003, p. 201-218.

MOURAD, G. *La segmentation de textes par l'étude de la ponctuation*, Acte de colloque international, CIDE'99. Damas, 1999, p. 155-171.

RIEGEL, Martin, PELLAT, Jean-Christophe, RIOUL, René, *Grammaire méthodique du français,* Paris, PUF, 1994.